

Terrain Adaptive Navigation for Planetary Rovers

Daniel Helmick

Mobility and Manipulation Group

Jet Propulsion Laboratory

California Institute of Technology

Pasadena, CA 91109

`dmh@jpl.nasa.gov`

Anelia Angelova

Department of Computer Science

California Institute of Technology

Pasadena, CA 91125

`anelia@vision.caltech.edu`

Larry Matthies

Computer Vision Group

Jet Propulsion Laboratory

California Institute of Technology

Pasadena, CA 91109

`lhm@jpl.nasa.gov`

Abstract

This paper describes the design, implementation, and experimental results of a navigation system for planetary rovers called Terrain Adaptive Navigation (TANav). This system was designed to enable greater access to and more robust operations within terrains of widely varying slippage. The system achieves this goal by using onboard stereo cameras to remotely classify surrounding terrain, predict the slippage of that terrain, and use this informa-

tion in the planning of a path to the goal. This navigation system consists of several integrated techniques: goodness map generation, terrain triage, terrain classification, remote slip prediction, path planning, high-fidelity traversability analysis (HFTA), and slip-compensated path following. Results from experiments with an end-to-end onboard implementation of the TANav system in a Mars analog environment are shown and compared to results from experiments with a more traditional navigation system that does not account for terrain properties.

1 Introduction

In this paper a navigation system designed for planetary rovers operating within terrains of potentially widely varying slippage (see Figure 1) is described. The current navigation system running on the Mars Exploration Rovers (MER) (Goldberg et al., 2002), (Maimone and Biesiadecki, 2006), was designed for relatively benign terrains and does not explicitly account for terrain types or potential slippage when evaluating or executing paths. The Terrain Adaptive Navigation (TANav) system is able to estimate the terra-mechanical properties the surrounding terrain, plan a path that accounts for the vehicle response to this terrain, and robustly reach the desired goal.

It is well recognized that many scientifically interesting sites on Mars are in very rough terrains with the potential for significant slippage and the danger of the rover being immobilized by terrain hazards, possibly a mission ending scenario (Grant et al., 2006). The “follow the water” strategy taken by NASA’s Mars Exploration Program inherently requires



Figure 1: The Erebus panorama (MER Opportunity rover) shows terrains of potentially widely varying slippage.

access to demanding terrains such as dry river channel systems, putative shorelines, and gullies emanating from canyon walls. The TANav system is designed to greatly increase access to and enable more robust operations in this type of terrain. It is important that the next generation of Mars rovers have the capability to autonomously navigate through these terrains, not only to increase the science return efficiency, but also to enable access to previously inaccessible science sites.

The TANav system is designed to deal with these scenarios using more sophisticated (and thus more computationally expensive) terrain analysis; however, this system is also designed to converge to computational complexity similar to that of currently deployed navigation systems when the terrain is benign. This navigation system consists of several technologies that have been developed, integrated, and tested onboard a research rover in Mars analog terrains (see Figure 2). These technologies include: goodness maps generation, terrain triage, terrain classification, remote slip prediction, path planning, high-fidelity traversability analysis (HFTA), and slip-compensated path following.

For the purposes of this paper a geometric hazard is defined as any object or terrain feature that can be deemed to be non-traversable based entirely on knowledge of that object's geometry (e.g. as measured by a range sensor) and a non-geometric hazard is defined as any terrain that, by virtue of its physical properties (e.g. density, cohesion, internal friction angle, etc.) can hinder or stop entirely the motion of the rover (e.g. unconsolidated sand).

As a poignant example of how a non-geometric hazard can potentially cause a mission ending scenario, the MER Opportunity rover on sol 446 became immobilized by a sand dune in Meridiani Planum and nearly got permanently stuck (see Figure 3) (it was successfully extracted 38 sols later). The TANav system is designed to autonomously prevent this type of situation by determining that the terrain is hazardous and planning around it.



Figure 2: Pluto rover in the JPL Mars Yard

Section 2 describes research that is related to this paper and how the TANav system is

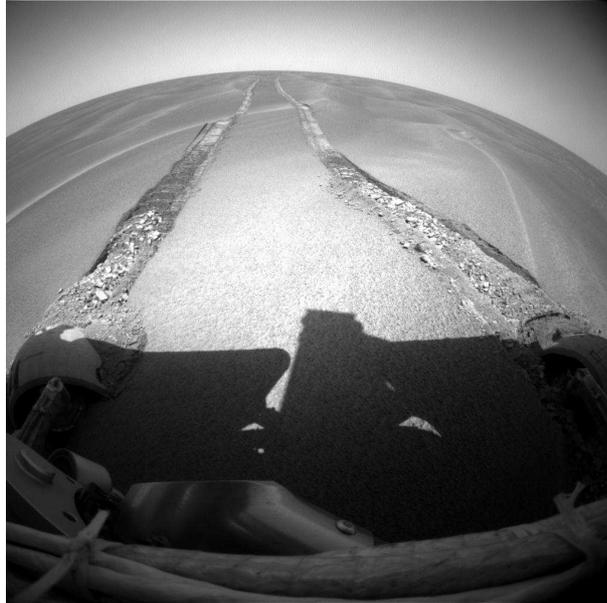


Figure 3: MER Opportunity rover stuck in a sand dune

distinguished from these bodies of work.

Section 3 discusses the system architecture as a whole, including design goals and operational assumptions. It explains how the subsystems are combined to create a navigation system. It also discusses the interfaces between the component technologies.

Section 3.2 describes the generation of goodness maps from stereo imagery (Goldberg et al., 2002), (Maimone and Biesiadecki, 2006). Also described in this section is the terrain triage algorithm. This is a technique to sub-divide the terrain into three categories based on the goodness calculation. The three categories are: *definitely traversable*, *definitely not traversable*, and *uncertain*. This categorization is then used to determine which parts of the future planned path need to be analyzed in more detail. The concept of terrain triage is the core idea that enables the computational complexity of the TANav system to scale with terrain difficulty and to converge to the computational complexity of traditional navigation

systems in benign terrain.

Section 3.3 discusses the remote slip prediction algorithm, which uses intensity and range data from stereo cameras to predict the slip of the rover on terrain at a distance (Angelova et al., 2007). It implements learned, non-linear regression models that output rover slip, using terrain geometry from stereo imagery as input. This predicted rover slip is then combined with the goodness map to create a “slip-augmented” goodness map. This section also discusses the role of a terrain classifier in slip prediction.

Section 3.5 briefly describes the path planner. The planner uses the D* algorithm (Stentz, 2005) to determine an optimal path to a goal through a cost map. It is beyond the scope of this paper to discuss this algorithm in detail, and the focus of this section is how the path planner fits into the overall system.

Section 3.6 explains the high-fidelity traversability analysis (HFTA) algorithm, which uses a sophisticated kinematic and dynamic forward simulation (Jain et al., 2004) of the rover following a path. It is designed to calculate a more accurate and realistic cost to traverse that path. The simulation includes a detailed geometric and mass model of the rover, terrain geometry generated from onboard stereo imagery, terra-mechanical properties generated from the terrain classifier (which are used in the dynamic wheel-terrain interaction model), and the same slip-compensated path following algorithm that runs onboard the rover (see Section 3.7). Because the same closed-loop path following algorithm is run in the forward simulation and onboard the actual rover the results of the simulation are much less dependent upon model inaccuracies and uncertainties than an open-loop forward simulation would be. The results of the HFTA are used to refine the parts of the planned path that go through

the *uncertain* regions of the triage map.

Once the final planned path is created, the slip-compensated path following algorithm (see Section 3.7) is invoked to enable the rover to actually follow this path regardless of the slip (Helmick et al., 2006). This algorithm compares visual odometry (VO) (a technique that measures rover motion including slippage) (Matthies and Schafer, 1987), (Maimone et al., 2007) and vehicle kinematics (a technique that measures rover motion not including slippage) (Helmick et al., 2006) to estimate the location and the slippage of the rover; it then compensates for this slippage and accurately follows the desired path to the goal.

Section 4 describes in detail the three major components of the experiments performed: the rover hardware, onboard software, and the Mars analog terrain.

Section 5 shows results from experiments with an end-to-end onboard implementation of the TANav system in a Mars analog environment and compares these results to those from experiments with a more traditional navigation system that does not account for terrain properties.

2 Related Work

A large amount of work has been done in the area of robot navigation in natural terrains. However, this paper is the first work to use learned, non-geometric terrain properties to enable planning through terrains of widely varying slippage. The navigation system of the MER Rovers (GESTALT) was designed for fairly benign terrain and does not account for non-geometric hazards (Goldberg et al., 2002), (Maimone and Biesiadecki, 2006). The use of stereo imagery to generate geometric goodness maps has been done by (Goldberg et al.,

2002), (Singh et al., 2000). Using a grid-based planner to generate optimal paths through cost maps has also been shown by (Stentz, 2005), (Carsten et al., 2007). The combination of geometric goodness maps and a grid-based planner, as is demonstrated in (Carsten et al., 2007), is very similar to the technique used in our control experiment so that our experimental results clearly distinguish the advantages of the TANav system over this approach.

Other work that uses the estimation of terra-mechanical properties to assess traversability of the terrain includes (Ishigami et al., 2006), (Iagnemma et al., 2002), (Ojeda et al., 2006). (Ishigami et al., 2006) uses analytical models of a single wheel interacting with the terrain to infer a vehicle-level response to the terrain. (Iagnemma et al., 2002) uses analytical models of a single wheel interacting with the terrain in order to estimate parameters of the terrain, but they do not suggest how these parameters could be used for traversability analysis or path planning. Similarly, (Ojeda et al., 2006) uses onboard sensors to classify terrain and characterize “traficability”, but they also do not address how this can be used to improve navigation.

The use of terrain classification and characterization for navigation purposes has been done by (Manduchi et al., 2005), (Halatci et al., 2008), (Talukder et al., 2002), (Dima et al., 2004). (Halatci et al., 2008) specifically addresses the challenges of terrain classification for planetary rovers. None of these papers, however, specifically address the integration of the terrain classification with a path planner. (Bajracharya et al., 2008) does specifically address this issue for terrestrial vehicles, however, it is done by classifying terrain into only two classes, “traversable” and “not traversable” which is not very applicable to planning optimal paths on planetary surfaces.

The use of forward simulation/modeling of the vehicle in planning has been done by (LaValle and Ku, 1999), (Melchior and Simmons, 2007), (Howard and Kelly, 2007), (Howard et al., 2008), however, the explicit use of stereo-generated terrain geometries, remotely learned terra-mechanical properties, and a closed-loop path following algorithm integrated with this forward simulation (as is done by the HFTA algorithm) has never been done before.

3 System Architecture

3.1 Summary

Figure 4 is a block diagram showing the system architecture described in this paper. The different colors represent functional groups of this architecture: red represents sensing, green represents mapping and terrain analysis, yellow represents path planning, and blue represents path following.

The top block is navigation camera (Navcam) imagery. The Navcams on the research rovers (Kim et al., 2005), the MER rovers (Maki et al., 2003), and the MSL (Malin et al., 2005) rover are all stereo cameras mounted on pan/tilt masts. These cameras allow the rover to take panoramic images from a high perspective, which decreases obscurations, thus enabling terrain sensing at further distances. Typical camera configurations are shown in Table 1. As can be seen in this table, some of these configurations allow for accurate stereo ranging at distances up to 50 meters. With the pan/tilt capability, range information spanning 360° can be accurately registered into a single map. The accuracy of this registration can be derived from the pointing accuracy of the pan/tilt mechanism, which on MER was 1 mrad (Warden et al., 2004), which corresponds to 5 cm registration error at 50 meters.

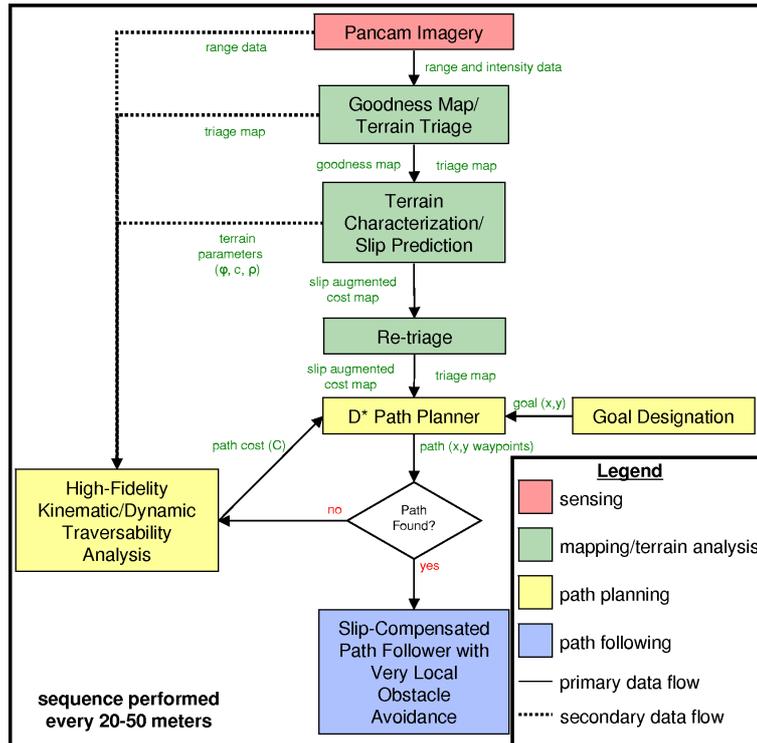


Figure 4: System architecture block diagram

Using the MSL Pancams, stereo errors of less than 20 cm at 50 meters range are possible. Other techniques such as wide-baseline stereo could increase the range and the accuracy even further (Olson et al., 2003), (Olson and Abi-Rached, 2005).

The assumed operational scenario of this navigation architecture is that a goal is designated within stereo range of the rover. The maximum distance for goal designation is a function of stereo range error at that distance, vehicle pose estimation error at that distance, and acceptable path error. Given the MSL Pancam range error (see Table 1) and the fact that pose error is 1-2% of distance traveled (Helmick et al., 2006), it is feasible that goals of up to 50 meters away would be acceptable, but more conservative goals of 20 to 30 meters away are more likely.

Table 1: MER/MSL Navcam and Pancam configurations (range error assumes 0.25 pixel stereo correlation accuracy)

	MER/MSL Navcams	MER Pancams	MSL Pancams
baseline (meters)	0.20	0.30	0.20
camera resolution (pixels)	1024×1024	1024×1024	1200×1200
pixel size (microns)	12	12	7.4
camera field of view (FOV)	45.0° × 45.0°	16.0° × 16.0°	5.1° × 5.1°
range error at 50 meters (meters)	2.5	0.60	0.19

Once a goal is designated, a stereo panorama is taken in the general direction of the goal (if a panorama has not already been taken for goal designation). Stereo is then done on each pair of the images and registered into a map using the pan/tilt angles. Once the stereo point cloud data are registered, a goodness map and a triage map are generated using planar statistics on the terrain geometry (as described in Section 3.2). The goodness map is then augmented with slip prediction costs (Section 3.3), re-triaged, and then passed to the path planner. The path planner plans an optimal path from the current rover location to the designated goal (Section 3.5). Because the path planner is performed on the slip-augmented goodness map, the resulting path avoids both geometric and non-geometric hazards. If all of the planned path goes through *definitely traversable* terrain, then the path is passed to the slip-compensated path follower. If any part of the path goes through *uncertain* regions of the triage map then the HFTA is performed on those regions to obtain a more realistic cost of traversal and to refine the path in that region (see Section 3.6). The refined path is then passed to the path follower and the rover follows the path until it reaches the goal (see Section 3.7).

As mentioned in the introduction, one of the goals in the design of this architecture was

to create a rover navigation system that would be capable of navigating through rough, high-slip terrains, but which would converge to the computational complexity of simpler algorithms in benign terrain. This is enabled using the terrain triage algorithm described in Section 3.2.

3.2 Goodness Map/Terrain Triage

A goodness map is a regularly spaced grid representing the local region around the rover. The map is populated using stereo data generated from panoramic imagery taken from the mast. In each cell of the map, a goodness value is calculated using the stereo data that falls in and around that cell. A plane is then fit to a rover sized patch of cells. The goodness calculation involves metrics such as pitch and roll of the plane, roughness of the terrain (standard deviation of the plane fit), and step heights within the patch (Goldberg et al., 2002).

Terrain triage is a simple concept that is fundamental to reducing the computational complexity of this navigation architecture in benign terrains. The idea is to categorize the terrain into three categories: *definitely traversable*, *definitely not traversable*, and *uncertain*. This categorization is done with an inexpensive algorithm that thresholds each cell of the goodness map by its geometric goodness value, thus binning each of the cells into one of the three categories. This step is performed twice: once before the slip prediction and once after. The first time is to determine whether or not slip prediction needs to be performed for a particular cell in the map. If a cell is categorized as *definitely not traversable* then there is no need to do slip prediction for that cell, because the slip prediction algorithm can only increase the cost of the cell. The second time terrain triage is performed is to incorporate

the new costs from the slip prediction into the goodness map. Next, when a path is planned through the goodness map (as described in Section 3.5), if the planned path travels entirely through *definitely traversable* terrain, then this path is deemed acceptable and it is passed to the slip-compensated path follower without any further analysis. This will only happen in relatively benign terrain. If, however, the path passes through *uncertain* terrain, then the HFTA (see Section 3.6) is invoked on those sections of the path. This decision point is represented by the diamond “path found?” box of Figure 4.

3.3 Terrain Classification/Slip Prediction

An independent assessment of the terrain traversability is done in terms of rover slippage. Slip is a measure of the lack of progress or the lack of mobility of the rover on a certain terrain. It is defined as the difference between the commanded velocity and the actual achieved velocity in each degree-of-freedom (DOF) of the rover (Helmick et al., 2006) and is normalized by the commanded velocity (Wong, 1993).

Rover slippage has been recognized to be a significant limiting factor for the MER rovers while driving on steep slopes (Biesiadecki et al., 2005), (Leger et al., 2005). Knowing the amount of slip beforehand and being able to detect areas of large slip will prevent the rover from getting stuck in dangerous terrain and will enable more intelligent path planning. Slip prediction is needed in addition to an obstacle detection mechanism because an area of large slip is a non-geometric type of obstacle and cannot be detected with a standard obstacle avoidance algorithm such as GESTALT (Goldberg et al., 2002).

3.3.1 Main Method and Architecture

While detecting rover slippage is relatively straightforward (Helmick et al., 2006), (Reina et al., 2006), the main challenge here is that the rover slip needs to be known remotely, before the rover actually traverses a particular location, in order to enable safe avoidance of areas of large slip. We have proposed an algorithm which infers the amount of slip on the upcoming terrain using visual information and onboard sensors (e.g. an IMU) and have shown successful slip prediction results from only these remote sensors. The problem is approached by learning from previous examples. To tackle the problem of slip prediction from a distance, we subdivide it into first recognizing the soil type the rover is going to traverse and then predicting the amount of slip as a function of terrain geometry, i.e. slopes (Angelova et al., 2007).

The main architecture of the slip prediction algorithm is given in Figure 5. For clarity, we first describe the prediction part of the algorithm, assuming the terrain classifier and the slip models have already been learned. The slip prediction module receives stereo pair imagery and rover attitude with respect to gravity as inputs. A map of the environment is built using the stereo range data registered with the color and texture information from the input images. In particular, each cell of the map contains information about terrain elevation and points to an image patch which has observed this cell. To predict slip in a map cell, the terrain classifier is applied to all the map cells in its neighborhood. A majority voting among their responses is used as the final terrain classification response at the desired rover location. Then, a locally linear fit in the cell's neighborhood is performed to retrieve the local slope under the potential rover footprint. The slopes are decomposed into a longitudinal and a

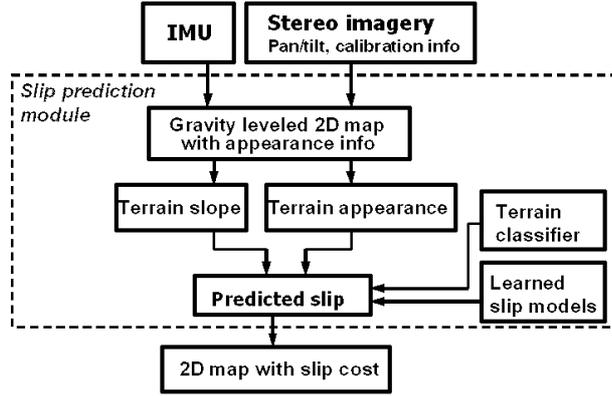


Figure 5: Slip prediction algorithm framework

lateral slope with respect to the orientation of the rover. The two slope angles are used as inputs to a pre-learned nonlinear slip model for the particular terrain type determined by the terrain classification algorithm. The output of this module is the predicted slip for a given orientation of the rover and a slip related cost at a given location. The slip related cost is an estimate of rover mobility without regards to particular robot orientation. It is used by the path planning algorithm by producing a slip goodness map. This map is generated by linearly combining the predicted longitudinal and lateral slip for each cell of the map and selecting the maximum slip over a range of rover yaw.

During training, the rover collects appearance and geometry information about a particular location while it is observed by the rover from a distance. The corresponding slip of the rover is also measured when this location is being traversed. We use visual odometry (VO) between two consecutive steps to estimate the actual rover velocity. The commanded velocity of the rover is computed by using the full vehicle kinematics. The collected data pairs of visual information and slip measurements are given to the learning module which learns a terrain type classifier and independent slip models for each terrain type (Angelova et al.,

2007). More details of the two training components are given in the next two subsections. A rover position estimation is computed within the slip prediction module by accumulating the VO estimates. This is necessary to be able to map the current rover location to a location previously observed by the rover from a distance.

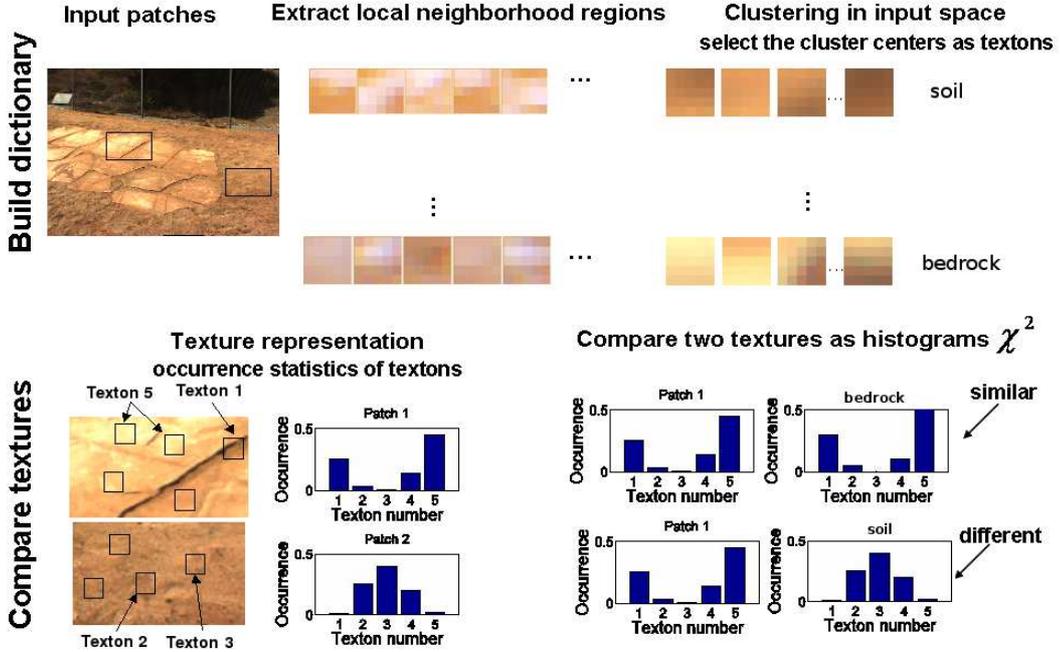


Figure 6: Terrain classification diagram

3.3.2 Terrain Classification

Our approach to terrain classification is based on processing visual appearance information, namely texture and color. We apply the texton based algorithm proposed in (Varma and Zisserman, 2003) which uses both color and texture simultaneously to learn to discriminate different terrain appearance patches. The algorithm proceeds as follows: initially the color R,G,B values in small pixel neighborhoods are collected and the most frequent features in the whole data are selected. Then a histogram of the occurrence of any of the selected features

within a patch corresponding to a map cell is built and compared by using a nearest neighbor classifier to a database of training patches (Varma and Zisserman, 2003). Intuitively, a patch from a bedrock class will have a high frequency of pixels typical of previously observed bedrock patches, but it might also contain a small number of pixels which are typical of an unfamiliar to the system “pebble” class which happens to be also shared with the soil class too (either of the terrain types patches might have small rocks dispersed in them). That is, this representation allows for building more complex appearance models and making correct decisions given the observed statistics from the data. In order to deal with variations in lighting conditions this algorithm uses normalized color during the training and classification steps. This approach has worked successfully in the past (Manduchi et al., 2005) and was effective in this research, although there is still work that could be done to make it more robust, as mentioned in Section 7. The training data and the data from the test runs (see Section 5.3) were taken under a wide range of lighting conditions and the classification was still quite effective.

Evaluation of the terrain classifier for this particular data domain has been provided in (Angelova et al., 2006). The terrain classification results are satisfactory and give successful slip prediction results (Angelova et al., 2007). The appearance-based terrain classifier can be improved by adding more sensors, both visual (e.g. multi-spectral imagery) or mechanical (e.g. vehicle vibrations).

Apart from being an instrumental part to the slip prediction module, the terrain classification provides important information to the HFTA algorithm (see Section 3.6). After the terrain type has been recognized, a canonical set of soil parameters associated with it are passed to the kinematic and dynamic simulation of the rover on the part of the terrain which has been

deemed *uncertain*.

3.3.3 Learning the Slip Models

As each terrain type has a potentially different slip behavior (Bekker, 1969), (Wong, 1993), we learn a slip model for each terrain independently. The slip models are built by learning a nonlinear approximation function which maps terrain slopes to the measured slip. The goal is to learn slip as a function of the terrain slopes: $S = S(x_{longit.}, x_{lateral})$. This function is known to be highly nonlinear (Lindemann and Voorhees, 2005) and so we learn the slip by applying a receptive field regression technique (Vijayakumar et al., 2005). This technique separates the input domains into subregions, called receptive fields, and applies locally linear fits to the data to approximate a globally nonlinear function.

3.3.4 Implementation Details

The software architecture of the algorithm is designed to provide efficient slip prediction. Because terrain classification from visual information is generally time consuming, the focus has been on decreasing the amount of computation devoted to image processing related to terrain classification. In particular, our main idea is to evaluate the terrain type per map cell, rather than evaluating the terrain type in the whole image. This design concept can give significant advantages. Some speed-up can be achieved, as parts of the image do not belong to the map (e.g. the pixels above the horizon). Additionally, the terrain classifier will not be invoked if slip prediction is not needed in a certain area, e.g. an area which the terrain triage has already marked as *definitely not traversable*. Finally, a map cell at a close range covers a large part of the image compared to the ones at far ranges and can be processed

selectively to speed up the processing without hurting the overall performance. To achieve that, the map cell structure we use saves only its 3D location and pointers to images which have observed it. When the terrain type needs to be predicted in a particular map cell, a projection of the map cell to the image is done and an image patch corresponding to this cell is retrieved.

Additionally, this paradigm allows for stereo imagery data to be received asynchronously or intermittently. It also enables more efficient computation when multiple overlapping images are acquired (as is typical with a panorama) by evaluating the terrain classification once per cell rather than once per image. The result of the terrain classification is saved with its corresponding confidence and might be combined with a potentially new evaluation if the confidence is insufficient. This is in contrast to processing fully all of the incoming images, extracting visual features and saving them to the map cells. See Figure 7 for a graphical representation of this concept. This paradigm fits with the rest of the system because the terrain triage algorithm provides its output as map cells, and thus there is no need to classify the image patch that corresponds to terrain that has already been deemed *definitely not traversable*.

3.4 Map Merging

The geometric goodness map described in Section 3.2 can then be merged with the slip goodness map described in Section 3.3.1. This merge is performed on a cell-by-cell basis using a weighted linear combination of each cell goodness. This “slip-augmented” goodness map is then used by the path planner described in Section 3.5. It is important to note that the slip goodness map cannot increase the goodness of any cell of the geometric goodness

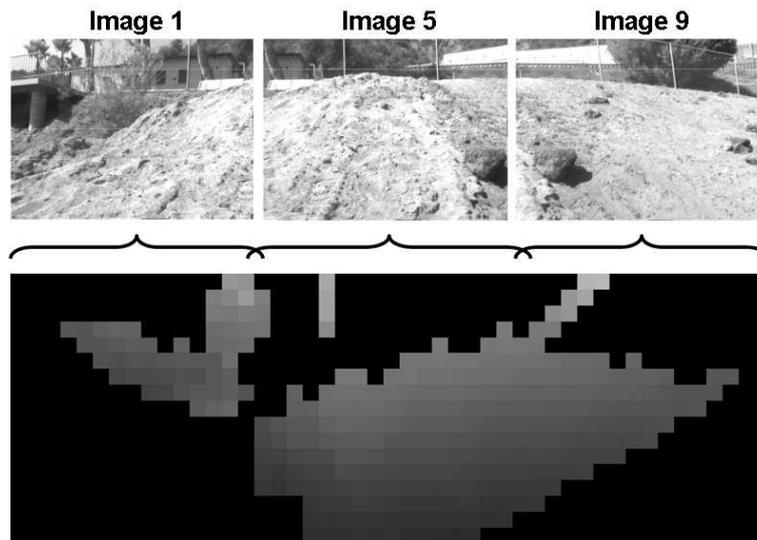


Figure 7: Schematic of the software design paradigm: each map cell keeps a pointer to an image which has observed it and the terrain classification is done only if needed. For example, the elevation map shown has been built from nine panoramic stereo image pairs, but effectively the visual information from only three images will need to be processed to fully classify the terrain.

map (and vice-versa) so that geometric and slip hazards are both preserved in the final slip-augmented goodness map.

3.5 Path Planning

Once the goodness map is populated with information derived from terrain geometry and the slip prediction algorithm as described above, a path can be planned through this map from any start to any goal. We use a standard implementation of the D* algorithm (Stentz, 2005) to plan this path. It is beyond the scope of this paper to go into the algorithmic details of this planner. In summary, it is a derivative of the well known A* search algorithm, with the capability to do efficient, incremental replanning.

3.6 High-Fidelity Traversability Analysis

If any section of the path goes through *uncertain* regions of the triage map, then the HFTA algorithm is invoked onboard the rover. HFTA is a full kinematic and dynamic forward simulation of the rover following a path. It is designed to find the lowest cost path through these *uncertain* regions.

The simulation infrastructure is provided by ROAMS (Jain et al., 2004). ROAMS is a kinematic/dynamic simulation for rovers interacting with terrain. A detailed geometric and mass model is used to represent the rover in the simulation. This model includes all 15 DOFs of the mobility system of the actual rover used in the experiments, Pluto (which is described in greater detail in Section 5). This includes the 12 active DOFs (six steering and six drive) and the 3 passive DOFs (rocker and two bogies). Each of the links connecting

these DOFs has a mass and center of gravity (CG). Each of the active DOFs has a dynamic model representing a motor, which can be commanded in the same way as the actual motors on the rover.

The terrain is modeled geometrically using a mesh. For HFTA, this mesh is generated from stereo data, so it represents a realistic geometric model of the terrain around the rover each time this algorithm is used. As with any stereo data from a single point of view, obscurations will occur that cause “range shadowing.” This is a well-known effect and the terrain model will simply linearly interpolate over these shadows, essentially creating a ramp between the top of object creating the shadow and the terrain visible on the far side of the object. This ramp is actually the worst case scenario of the obscured terrain (i.e. no terrain can possibly be higher than the ramp) so it is a conservative assumption.

Mechanical soil properties can also be associated with the terrain based on results from the terrain classification algorithm described in Section 3.3. The terrain is again gridded, with the capability of assigning to each cell independent values for the cohesion, friction angle, and density of the soil in that cell. This enables the representation of non-homogeneous terrain at arbitrary resolution. For the HFTA algorithm, the terrain classifier predicts the terrain type of each of the cells. Then, a canonical set of soil properties is associated with each of the terrain classes. The canonical soil properties used in these experiments were specific to the terrain in the Mars Yard (see Figures 2 and 9 and Section 4). Significant effort has been invested to determine these parameters for Martian soil (J. Bell, 2007), which would be used for a rover operating on Mars.

These mechanical soil properties are used in a dynamic wheel/soil interaction model used

to determine the rover sinkage and slippage (Sohl and Jain, 2005). This model calculates and resolves the 18 forces (3 at each of the 6 wheels) of the vehicle interaction with the environment. This results in an accurate calculation of the net motion of the rover, including sinkage and slippage, as well as other properties of the rover during the traverse such as wheel torques and wheel velocities.

A significant body of work on validation (comparison of simulation response to actual rover response) of the ROAMS simulation environment and its soil mechanics models are described in (Huntsberger et al., 2008).

For the forward simulation, the same slip-compensated path following algorithm that is used onboard to control the actual rover (Helmick et al., 2006), is used to control the simulated rover. So a path is passed to the simulation, the path-following algorithm follows the path over the sensor generated terrain that models the slippage of the simulated rover over this terrain. While this is being simulated, several metrics are being recorded that will enable the assessment of the traversability of that particular path. Because the same closed-loop path following algorithm is run in the forward simulation and onboard the actual rover the results of the simulation are much less dependent upon model inaccuracies and uncertainties than an open-loop forward simulation would be. Therefore the path that is evaluated in the HFTA algorithm is nearly identical to the path actually performed by the real vehicle. This increases the accuracy and reduces the uncertainty of the cost metrics described in the following paragraph.

The most important of these metrics is the energy required for the rover to traverse the given path. The first half of Equation 1 shows the energy calculation as the integral of the

product of wheel torque and wheel velocity. Because the wheel terrain interaction (and thus vehicle slip) is being modeled, and because the slip is being compensated for, this metric penalizes for both high slip terrain (because the wheels must turn for a longer period of time to reach the goal) and for rough terrain (because the wheel torques are higher on locally steeper terrain, i.e. rocks, gullies, etc.). The path cost is calculated using:

$$C = W_e \underbrace{\left(\sum_{n=1}^6 \int T_n \cdot \omega_n dt \right)}_{energy} + W_p \underbrace{\left(\int (\| p_{des} - p_{act} \|) dt \right)}_{path\ error} \quad (1)$$

where T_n and ω_n are the torque and speed of wheel n , respectively; p_{des} and p_{act} are the desired and actual rover positions, respectively; and W_e and W_p are weighting terms.

Another metric that is used is path error (the second half of Equation 1). Because the slip-compensated path follower is being run in the simulation, path error is an indication of terrain that is more difficult to traverse, even when compensating for slip.

Other metrics that can be used are: minimum ground clearance, risk of wheel trap, minimum distance to extreme hazards, and maximum ground interaction forces. Ground clearance is the distance from the ground to the underbelly of the rover. This is a potential hazard because it can “high-center” the rover which can be a very serious threat to mobility. This type of forward simulation is the only way to accurately estimate the minimum ground clearance of a path. A wheel trap is a specific type of obstacle that can get trapped between the wheels of the rover. It is a significant threat to mobility and can only be predicted with a high-fidelity forward simulation. Distance to extreme hazards (such as large drop-offs, high-slip areas, etc.) is another metric to determine the risk of traversing a path and can be used to evaluate the relative cost of a path. Large ground interaction forces that can be caused by traversing over rough, stiff terrain (such as very rocky terrain) could also be

used as a metric to increase the lifetime of the vehicle by selecting paths that place lower demands on the actuators and structure of the vehicle.

When the HFTA algorithm is invoked, the input is a single path through an *uncertain* region. Using the beginning and end points of the length of this part of the path, many paths are randomly generated using several simple heuristic rules (monotonic growth in x and y, limited distance per step, etc.) that provide sufficient coverage of the space between the beginning and end points. The simulation is then run on each of these paths, calculating a single scalar value of cost for each of the paths. It then outputs the path with the lowest cost.

3.7 Slip-Compensated Path Following

When the final path is created, the slip-compensated path following algorithm is invoked to enable the rover to actually follow this path regardless of the slip. This algorithm compares visual odometry (a technique that measures rover motion including slippage) and vehicle kinematics (a technique that measures rover motion not including slippage) to estimate the location and the slippage of the rover; it then compensates for this slippage and accurately follows the desired path to the goal. This system is described in detail in (Helmick et al., 2006).

4 Experimental Design

The experimental design consists of three major components: the rover hardware (see Figure 8), onboard software, and the Mars analog terrain (see Figures 2 and 9). The rover

hardware was designed to allow for rapid software development and convenient, practical usage while still remaining representative of flight hardware that has been used on previous missions and will be used on future planetary missions. Mechanically, it is a six-wheeled rover with a rocker/bogie suspension that is kinematically very similar to Sojourner, MER, and MSL (Lindemann and Voorhees, 2005) (Muirhead, 2004). The total mass of the rover is approximately 60 kg, the wheel diameter is 20 cm, the wheel width is 10 cm, the wheelbase is 75 cm, and the track width is 62 cm. The maximum linear velocity of the rover is 9 cm/sec and the maximum rotational velocity of the vehicle is 0.24 rad/sec. The onboard avionics consist of a standard PC-104 stack with a 1.8GHz Pentium-M processor, commercial-off-the-shelf (COTS) distributed motor controllers, fourteen brushless motors to actuate six steerable/drivable wheels and a pan/tilt mast, and eight cameras (six monochrome and two color). The cameras are configured to make four stereo pairs consisting of front and rear hazard avoidance camera pairs (Hazcams), and navigation (Navcams) and panorama (Pancams) camera pairs on the mast.

The onboard software runs as several separate processes on a standard Debian Linux distribution. One process is dedicated to coordinated motor commanding, kinematics, and slip-compensated path following and runs at 20 Hz. Another process is dedicated to image capture, stereo processing, and visual odometry and runs at 1 Hz. Several other processes are used during each planning cycle which occurs every 10-50 meters of traversal and consists of goodness map generation, terrain classification, slip prediction, path planning, and HFTA.

The Mars analog terrain is a widely used facility at JPL called the Mars Yard (see Figures 2 and 9). It is a large (55m x 45m) area consisting of multiple terrain types (various rock sizes and distributions, multiple sand/soil types, and several areas of bedrock simulant) and vari-



Figure 8: Pluto research rover

ous terrain geometries (including flat ground and slopes up to 30°). It was designed to have areas of widely varying slippage, analogous to Martian terrain, specifically to demonstrate and test mobility and navigation systems capable of dealing with such terrain.

5 Experimental Results

Results from experiments onboard a research rover in the JPL Mars yard are shown here. The main goal of the experiments was to show the difference in performance between the TANav system and a more traditional navigation system that only uses terrain geometry and does not account for terrain properties in its path planning. The comparison of these

two systems demonstrates the advantages that can be gained by using the TANav system. Both systems were started with the same stereo panorama (see Figure 9) from the mast from the same location in the Mars yard. In both cases a geometric goodness map was generated from this panorama. For the TANav experiment the local terrain was classified, and slip was estimated for the surrounding area. Then a slip-augmented goodness map was generated and a path was planned through this map. Next, terrain triage and the HFTA algorithm were performed to refine this path. For the control experiment a path was planned through the geometric goodness map. Finally, for both experiments, the rover performed slip-compensated path following on the corresponding path. Results from each of these steps are described below.

5.1 Stereo Panorama

The experiments began with the rover taking a panorama of the surrounding area with its panoramic cameras. This panorama consisted of five stereo pair color images each with approximately 20% overlap between images (see Figure 9). From this panorama, a user designated the desired goal of the traverse in the rover frame (x is forward and y is to the right) (see Figure 10). In this case the goal was chosen to be at $x=9.2$ and $y=2.6$ meters.

5.2 Goodness Map

Five point clouds were generated (one from each stereo pair of the panorama) using the stereo algorithm described in (Goldberg et al., 2002). These point clouds were merged and transformed to the rover frame using the pan/tilt angles of each of the image pairs. This merged point cloud was then used to generate a goodness map (using the algorithm described



Figure 9: Color panorama of Mars Yard taken by Pluto panoramic cameras (flat terrain on the left and slopes reaching 25° on the right)

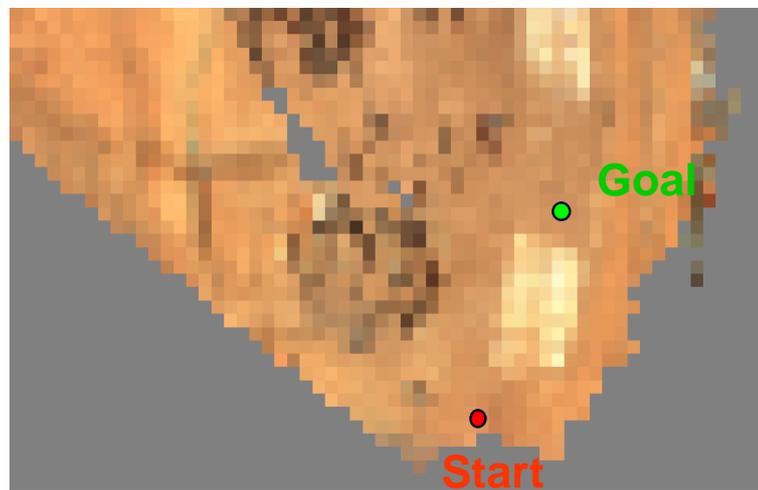


Figure 10: Overhead projection of panorama with start and goal points

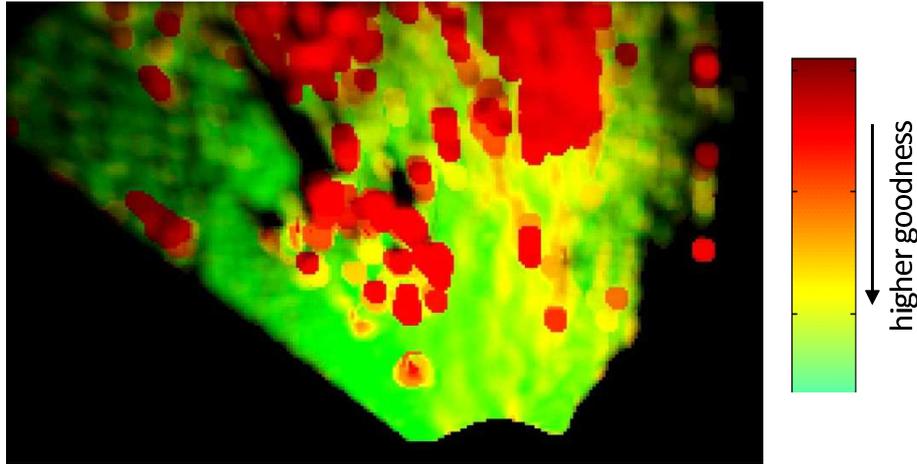


Figure 11: Geometric goodness map

in Section 3.2 and (Maimone and Biesiadecki, 2006)) with 10 cm grid cells covering an area 15 x 30 meters in front of the rover (see Figure 11). As can be seen in the goodness map, geometric hazards (the rock field to the left and the steep slopes to the right as can be seen in Figure 9) are accounted for in this map with lower goodness (red) cells and the flatter, more benign areas without rocks are shown with higher goodness (green).

5.3 Terrain Classification

Terrain classification was then performed onboard the rover on the panoramic imagery using the algorithm described in Section 3 and (Angelova et al., 2007). The results of this terrain classification are shown in Figures 12 and 13. As can be seen, the classifier was trained using three terrain classes: bedrock (blue), soil (red), and rock (green). This training was done using different images from different perspectives of the Mars Yard. There are two large patches of bedrock on the right hand side that have accurately been classified (see the right hand side of Figure 12 for an image space overlay) and the patches of rock on the left are also accurately classified.



Figure 12: Overlay of terrain classification results

The classification accuracy was calculated using two different metrics: N_c/N_{tc} and N_p/N_{tp} , where N_c and N_p are the number of terrain cells and pixels, respectively, where the classification result matched the hand-labeled class, and N_{tc} and N_{tp} are the total number of hand-labeled terrain cells and pixels, respectively. Each terrain cell is 20 x 20 cm. The classification accuracy for the entire panorama for the cell and pixel metrics are 84.4% and 88.1% respectively.

5.4 Slip Prediction

Slip prediction was then performed onboard the rover. Slip models for the three terrain classes were learned offboard using training data from many traverses over the Mars Yard terrain. Using these learned models and the results of the terrain classification and the stereo range data (to remotely determine terrain slopes) as inputs, a slip goodness map was generated (see Figure 14). This slip goodness map predicts a normalized value of rover slip (where lower goodness means higher slip and vice-versa) for each grid in the map for which there is range and terrain class data available.

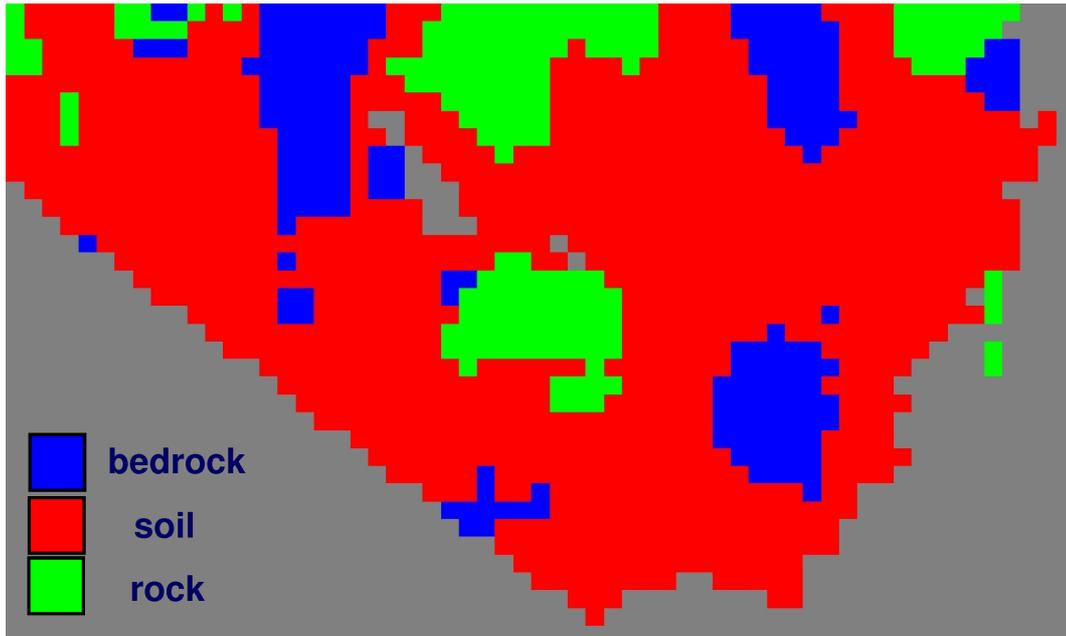


Figure 13: Overhead map of terrain classification results

As can be seen in the slip goodness map, the lower sloped terrain on the left (corresponding to both the rocks and the soil) shows higher goodness values (lower slip) and the higher slope terrain on the right (corresponding to the areas of soil) show lower goodness values (higher slip). The two areas of bedrock on the right (in the near and far field) show distinctly higher goodness even though the slopes are high (20-25°). This corresponds to the learned models of bedrock which exhibit extremely small slip responses even for high slopes.

5.5 Slip-Augmented Goodness Map

A slip-augmented goodness map (see Figure 15) was generated by merging the slip goodness map and the geometric goodness map as described in Section 3.4.

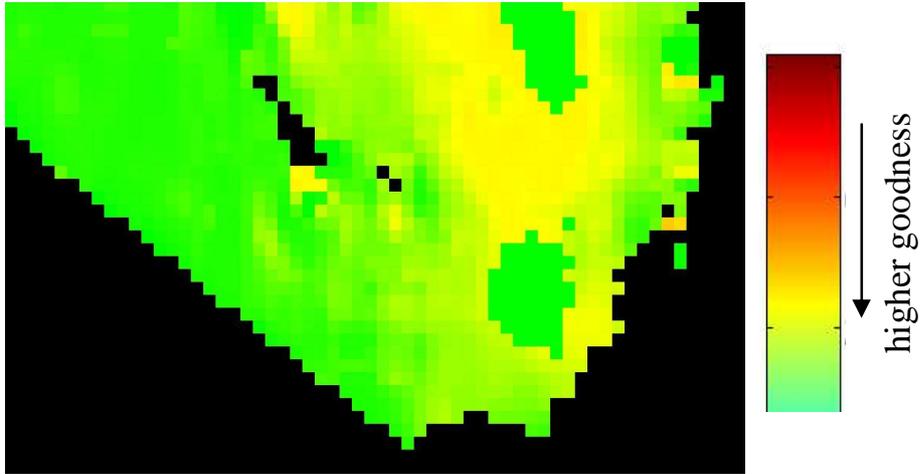


Figure 14: Slip goodness map

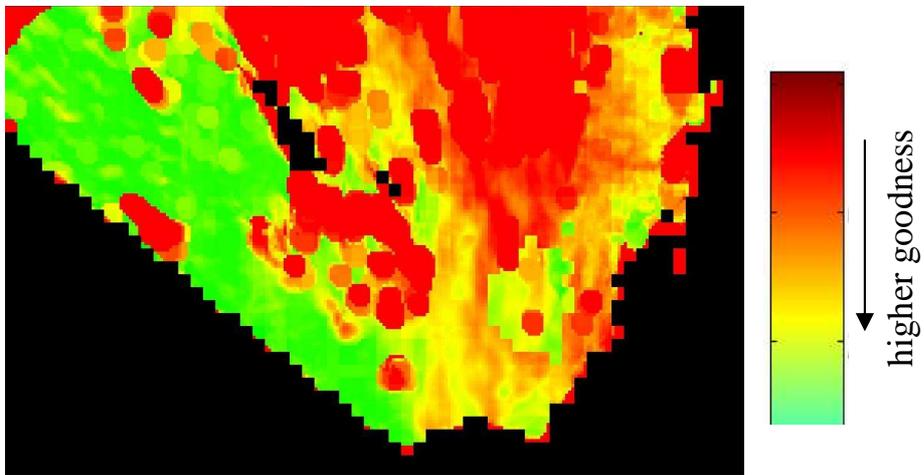


Figure 15: Slip-augmented goodness map

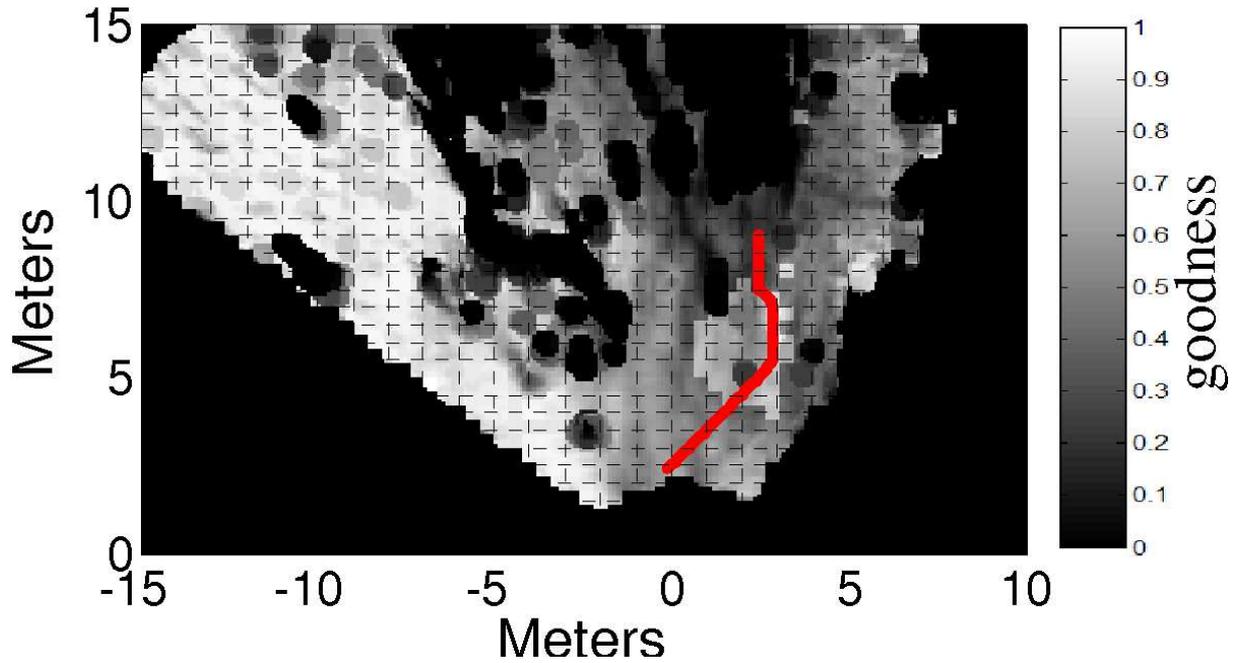


Figure 16: Planned path with TANav

5.6 Path Planning

Given the slip-augmented goodness map, the user designated goal, and the current location of the rover, an optimal path was planned for the TANav experiment (see Figure 16). For the control experiment, a path was planned using just the geometric goodness map (see Figure 17). As can be seen, the path using the geometric goodness map goes through the lower slope terrain on the left before curving up to the goal at the end of the path. The path using the slip-augmented goodness map takes a higher slope, lower slip path to the right, over the bedrock, and then curves down to the goal at the end of the path.

5.7 Terrain Triage

The terrain triage algorithm was then applied to the slip-augmented goodness map to create a triage map (see Figure 18). The part of the path circled in blue passes through an *uncertain*

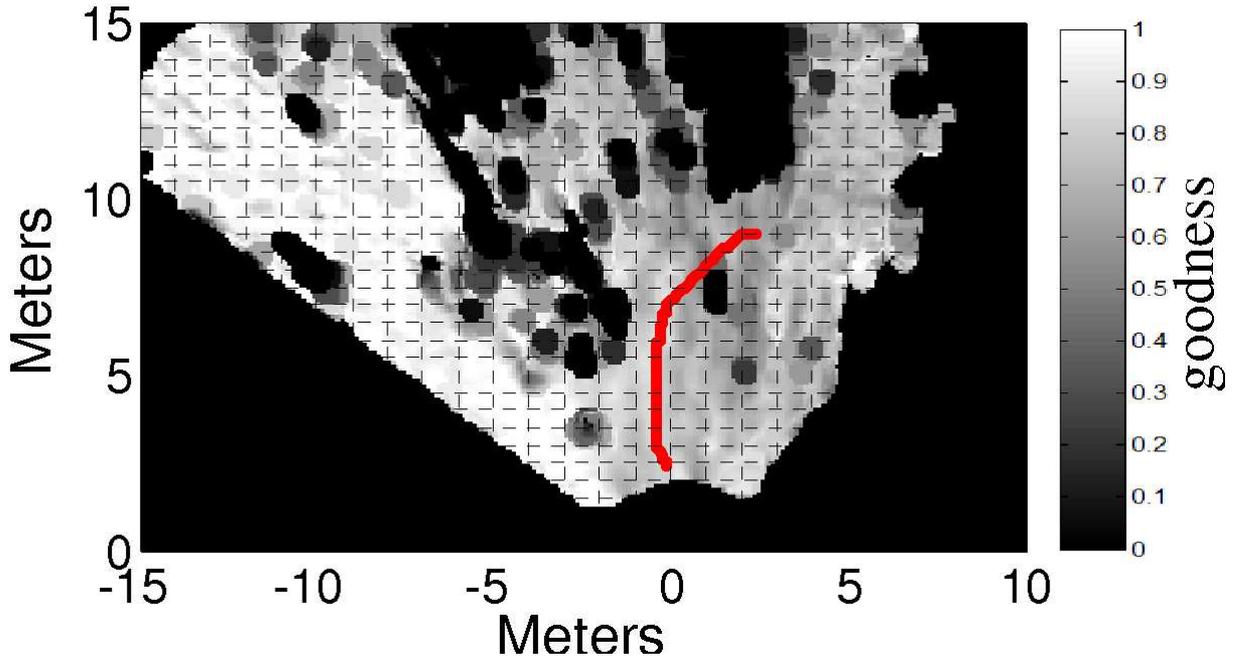


Figure 17: Planned path without TANav

region of the triage map, therefore this length of the path was used as an input to the HFTA algorithm.

5.8 High Fidelity Traversability Analysis

The HFTA algorithm, described in Section 3.6, was then invoked on the part of the path circled in blue in Figure 18. Fourteen randomly generated paths that were evaluated by the HFTA algorithm are shown in Figure 19. The cost of each of these paths as determined by the forward simulation is shown in Table 2. The stereo generated terrain that these paths were evaluated over is shown in Figure 20. In both the figure and the table, the lowest cost path (path “3”) is shown in bold red. The weights used in Table 2 were 1 and 10 for W_e and W_p respectively. These weights were determined with expert knowledge of the system, but it is important to note that the results of the HFTA algorithm are relatively insensitive to these

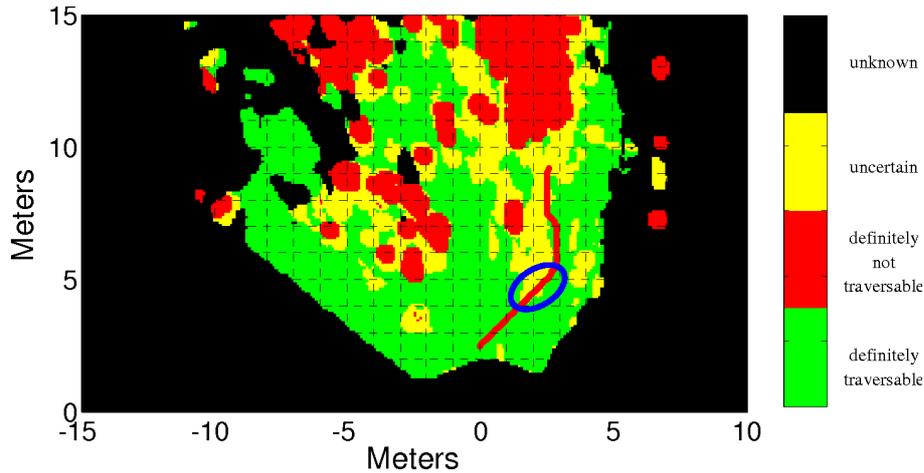


Figure 18: Terrain triage map

parameters. For example, path “3” remains the lowest cost path for all values of $W_p > \sim 5$. Path “3” was the final path sent to the slip-compensated path following algorithm. It is also worthwhile to note that the paths shown in Figure 19 are not the actual paths that the rover would traverse, they are the input paths to the slip-compensated path follower, which smooths the paths significantly. Because the slip-compensated algorithm is embedded in the HFTA algorithm, the path used to calculate the cost will be nearly identical to the path actually performed by the real vehicle.

5.9 Slip-Compensated Path Following

Each of the two paths generated above (with and without TANav) were followed by the slip-compensation path follower in the Mars Yard. The results from each of these traverses are shown in Figures 21 and 22. In each of these figures the red line represents the planned path, the green line represents the output from wheel odometry and the blue line represents the actual path traversed. The difference between wheel odometry and visual odometry is the slippage of the vehicle over the traverse. The traverse of the path using TANav experienced

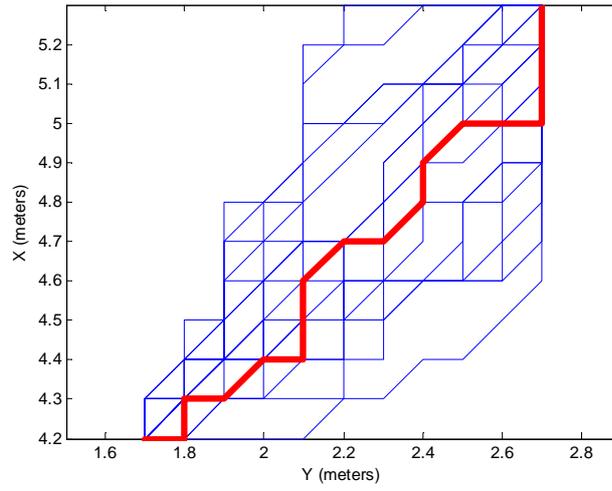


Figure 19: Randomly generated paths through *uncertain* region (area circled in Figure 18)

Table 2: HFTA Evaluated Path Costs

random path	energy (J)	path error (m-s)	cost
1	181.229	3.06218	211.8508
2	233.865	2.10263	254.8913
3	181.902	1.70714	198.9734
4	154.766	7.40014	228.7674
5	185.901	3.52733	221.1743
6	226.213	3.30995	259.3125
7	183.962	4.77294	231.6914
8	178.86	5.18531	230.7131
9	162.279	6.58719	228.1509
10	181.338	8.03311	261.6691
11	233.865	2.63038	260.1688
12	222.272	2.62189	248.4909
13	182.854	7.78614	260.7154
14	162.238	6.31429	225.3809

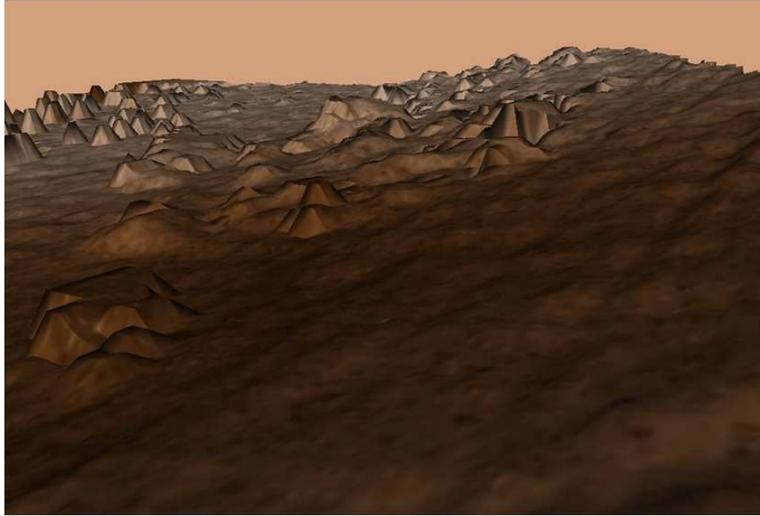


Figure 20: Stereo-generated HFTA terrain

a total of 1.78 meters of slippage (lateral and longitudinal) and successfully reached the goal. The traverse of the path that did not use TANav experienced 4.78 meters of slippage before the rover stalled due to excess slippage without ever having made it to the goal.

6 Conclusions

We have shown here the results from an onboard demonstration of the TANav system compared to those of a navigation system that does not use information about terrain properties. As can be seen from these results, the TANav system and the general concept of using slip estimation and terrain properties in a navigation system has the potential to significantly increase the efficiency, robustness, and safety of rover navigation in high-slip environments. The TANav system not only improved the performance of the system (by decreasing the total slip of the rover over the traverse) but it also enabled access to the goal by finding a path that was actually traversable. This is opposed to the system that did not use TANav, which resulted in a non-viable path that was not detected until traversal, which could potentially

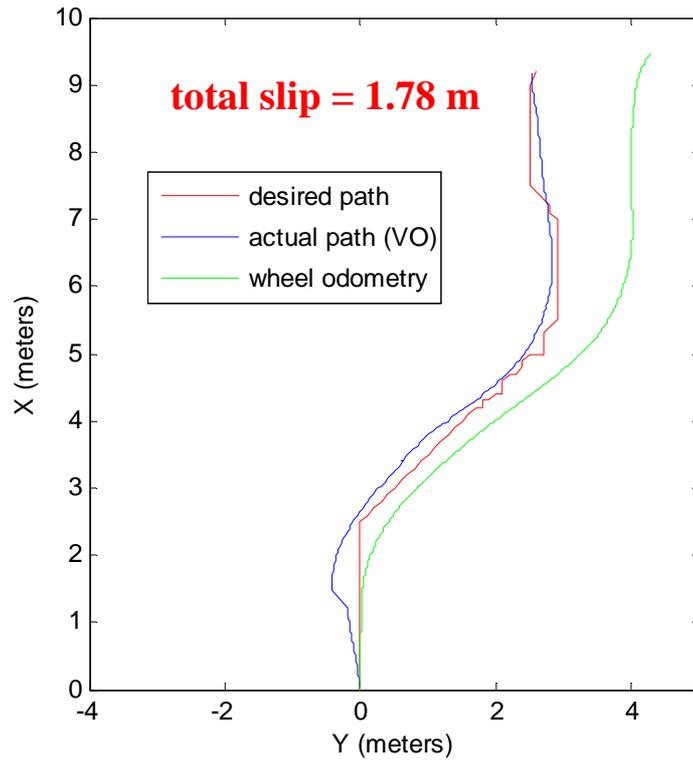


Figure 21: Path following results with TANav

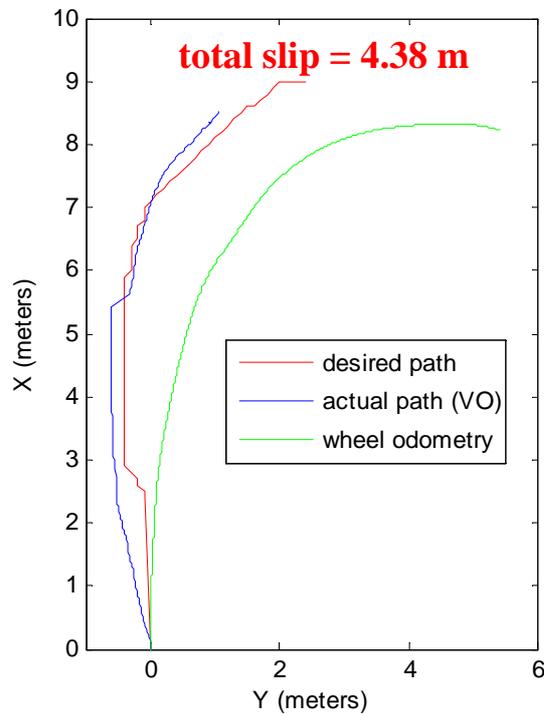


Figure 22: Path following results without TANav

result in a mission ending scenario.

7 Future Work

There are several areas of potential improvements that would be good next steps for this research. One of these would be to wrap the HFTA algorithm with a black box optimization algorithm such as Nelder-Mead (Nelder and Mead, 1964) in place of the random path generator. A second area of improvement would be to integrate directional slip prediction (which is currently done in the slip prediction step) into the path planning algorithm, which would enable higher-level behaviors to emerge such as planning a path with switch-backs up a slope. Another potential area for future work is dealing more completely with variations in lighting conditions in the terrain classifier. Finally, a statistical comparison of experiments involving the TANav system and traditional navigation systems would be a valuable contribution and would be important to quantitatively measure the improvements achieved by the TANav system.

8 Acknowledgements

The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology under contract from the National Aeronautics and Space Administration (NASA), with funding from the Mars Technology Program, NASA Science Mission Directorate.

References

- Angelova, A., Matthies, L., Helmick, D., and Perona, P. (2006). Slip prediction using visual information. In *Robotics: Science and Systems Conference*.
- Angelova, A., Matthies, L., Helmick, D., and Perona, P. (2007). Learning and prediction of slip from visual information. *Journal of Field Robotics*, 24(3):205–231.
- Bajracharya, M., Howard, A., Matthies, L., Tang, B., and Turmon, M. (2008). Autonomous Off-Road Navigation with End-to-End Learning for the LAGR Program. *Journal of Field Robotics*, Forthcoming.
- Bekker, M. (1969). *Introduction to Terrain-vehicle Systems*. Univ. of Michigan Press.
- Biesiadecki, J., Baumgartner, E., Bonitz, R., Cooper, B., Hartman, F., Leger, C., Maimone, M., and Maxwell, S. (2005). Mars Exploration Rover Surface Operations: Driving Opportunity at Meridiani Planum. In *IEEE Conference on Systems, Man, and Cybernetics*.
- Carsten, J., Rankin, A., Ferguson, D., and Stentz, A. (2007). Global Path Planning on Board the Mars Exploration Rovers. In *IEEE Aerospace Conference*, Big Sky, MT.
- Dima, C., Vandapel, N., and Hebert, M. (2004). Classifier Fusion for Outdoor Obstacle Detection. In *International Conference on Robotics and Automation*.
- Goldberg, S., Maimone, M., and L. Matthies (2002). Stereo Vision and Rover Navigation Software for Planetary Exploration. In *IEEE Aerospace Conference*, Big Sky, MT.
- Grant, J., Steele, A., Richardson, M., Bougher, S., Banerdt, B., Borg, L., Gruener, J., and Heldmann, J. (2006). Mars Science Goals, Objectives, Investigations, and Priorities: 2006. *Mars Exploration Program Analysis Group (MEPAG)*. <http://mepag.jpl.nasa.gov/reports/index.html>.

- Halatci, I., Brooks, C., and Iagnemma, K. (2008). A study of visual and tactile terrain classification and classifier fusion for planetary exploration rovers. *Robotica*, Forthcoming.
- Helmick, D. M., Roumeliotis, S. I., Cheng, Y., Clouse, D., Bajracharya, M., and Matthies, L. (2006). Slip-Compensated Path Following for Planetary Exploration Rovers. *Journal of Advanced Robotics*, 20(11):1257–1280.
- Howard, T., Green, C., Kelly, A., and Ferguson, D. (2008). State space sampling of feasible motions for high-performance mobile robot navigation in complex environments. *Journal of Field Robotics*, 25(6-7):325–345.
- Howard, T. and Kelly, A. (2007). Optimal Rough Terrain Trajectory Generation for Wheeled Mobile Robots. *International Journal of Robotics Research*, 26(2):141–166.
- Huntsberger, T., Jain, A., Cameron, J., Woodward, G., Myers, D., and Sohl, G. (2008). Characterization of the ROAMS Simulation Environment for Testing Rover Mobility on Sloped Terrain. In *9th International Symposium on Artificial Intelligence, Robotics and Automation in Space*, Universal City, CA.
- Iagnemma, K., Shibly, H., and Dubowsky, S. (2002). On-Line Traction Parameter Estimation for Planetary Rovers. In *IEEE International Conference on Robotics and Automation*.
- Ishigami, G., Miwa, A., Nagatani, K., and Yoshida, K. (2006). Terramechanics-based Analysis on Slope traversability for a Planetary Exploration Rover. In *International Symposium on Space Technology and Science*, Kanazawa, Japan.
- J. Bell, e. (2007). *The Martian Surface: Composition, Mineralogy, and Physical Properties*. Cambridge University Press.
- Jain, A., Balaram, J., Cameron, J., Guineau, J., Lim, C., Pomerantz, M., and Sohl, G.

- (2004). Recent Developments in the ROAMS Planetary Rover Simulation Environment. In *IEEE Aerospace Conference*, Big Sky, MT.
- Kim, W. S., Ansar, A. I., and Steele, R. D. (2005). Rover Mast Calibration, Exact Camera Pointing, and Camera Handoff for Visual Target Tracking. In *IEEE 12th Int. Conf. on Advanced Robotics*.
- LaValle, S. and Ku, J. (1999). Randomized kinodynamic planning. In *International Conference on Robotics and Automation*.
- Leger, C., Trebi-Ollennu, A., Wright, J., Maxwell, S., Bonitz, R., Biesiadecki, J., Hartman, F., Cooper, B., Baumgartner, E., and Maimone, M. (2005). Mars Exploration Rover Surface Operations: Driving Spirit at Gusev Crater. In *IEEE Conference on Systems, Man, and Cybernetics*.
- Lindemann, R. and Voorhees, C. (2005). Mars Exploration Rover mobility assembly design, test and performance. In *IEEE International Conference on Systems, Man and Cybernetics*.
- Maimone, M., Cheng, Y., and Matthies, L. (2007). Two years of Visual Odometry on the Mars Exploration Rovers: Field Reports. *Journal of Field Robotics*, 24(3):169–186.
- Maimone, M. W. and Biesiadecki, J. J. (2006). The Mars Exploration Rover Surface Mobility Flight Software: Driving Ambition. In *IEEE Aerospace Conference*, Big Sky, MT.
- Maki, J. N., III, J. F. B., Herkenhoff, K. E., Squyres, S. W., Kiely, A., Klimesh, M., Schwochert, M., Litwin, T., Willson, R., Johnson, A., Maimone, M., Baumgartner, E., Collins, A., Wadsworth, M., Elliot, S. T., Dingizian, A., Brown, D., Hagerott, E. C., Scherr, L., Deen, R., Alexander, D., and Lorre, J. (2003). The Mars Exploration Rover Engineering Cameras. *Journal Geophysical Research*, 108(12):1–24.

- Malin, M. C., Bell, J. F., Cameron, J., Dietrich, W. E., Edgett, K. S., Hallet, B., Herkenhoff, K. E., Lemmon, M. T., Parker, T. J., Sullivan, R. J., Sumner, D. Y., Thomas, P. C., Wohl, E. E., Ravine, M. A., Caplinger, M. A., and Maki, J. N. (2005). The Mast Cameras and Mars Descent Imager (MARDI) for the 2009 Mars Science Laboratory. In *36th Lunar and Planetary Science Conference*.
- Manduchi, R., Castano, A., Talukder, A., and Matthies, L. (2005). Obstacle Detection and Terrain Classification for Autonomous Off-road Navigation. *Autonomous Robots*, 18:81–102.
- Matthies, L. and Schafer, S. (1987). Error modeling in stereo navigation. *IEEE Journal of Robotics and Automation*, 3(3).
- Melchior, N. and Simmons, R. (2007). Particle RRT for Path Planning with Uncertainty. In *IEEE International Conference on Robotics and Automation*, Rome, Italy.
- Muirhead, B. K. (2004). Mars rovers, past and future. In *IEEE Aerospace Conference*, Big Sky, MT.
- Nelder, J. and Mead, R. (1964). A simplex method for function minimization. *The Computer Journal*, 7:308–313.
- Ojeda, L., Borenstein, J., Witus, G., and Karlsen, R. (2006). Terrain Characterization and Classification with a Mobile Robot. *Journal of Field Robotics*, 23(2):103–122.
- Olson, C. F. and Abi-Rached, H. (2005). Wide-Baseline Stereo Experiments in Natural Terrain. In *In Proceedings of the 12th International Conference on Advanced Robotics*.
- Olson, C. F., Abi-Rached, H., Ye, M., and Hendrich, J. P. (2003). Wide-Baseline Stereo Vision for Mars Rovers. In *IEEE International Conference on Intelligent Robots and Systems*.

- Reina, G., Ojeda, L., Milella, A., and Borenstein, J. (2006). Wheel slippage and sinkage detection for planetary rovers. *IEEE/ASME Transactions on Mechatronics*, 11(2):185–195.
- Singh, S., Simmons, R., Smith, T., Stentz, A., Verma, V., Yahja, A., and Schwehr, K. (2000). Recent Progress in Local and Global Traversability for Planetary Rovers. In *International Conference on Robotics and Automation*.
- Sohl, G. and Jain, A. (2005). Wheel-Terrain Contact Modeling in the Roams Planetary Rover Simulation. In *Fifth ASME International Conference on Multibody Systems, Nonlinear Dynamics and Control*, Long Beach, CA.
- Stentz, A. (2005). The Focussed D* Algorithm for Real-Time Replanning. In *Proceedings of IJCAI*.
- Talukder, A., Manduchi, R., Rankin, A., Owens, K., Matthies, L., Castano, A., and Hogg, R. (2002). Autonomous terrain characterization and modeling for dynamic control of unmanned vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland.
- Varma, M. and Zisserman, A. (2003). Texture classification: Are filter banks necessary? In *Conf. on Computer Vision and Pattern Recognition*.
- Vijayakumar, S., D’Souza, A., and Schaal, S. (2005). Incremental online learning in high dimensions. *Neural Computations*, 17(12):2602–2634.
- Warden, R., Cross, M., and Harvison, D. (2004). Pancam Mast Assembly on Mars Rover. In *37th Aerospace Mechanisms Symposium*, Johnson Space Center, Houston, TX.
- Wong, J. (1993). *Theory of Ground Vehicles*. John Wiley & Sons Inc.